Clase 01 - Prototipado y conceptos básicos de HTML

# Prototipado desde el papel

Para iniciar cualquier proyecto WEB es necesario tener un bosquejo de lo que queremos hacer. De esta manera **evitamos grandes cambios en las siguientes etapas** **del proyecto**. También nos permite definir de una manera rápida y sencilla los bloques o secciones que va a tener cada una de las páginas de un sitio web, de modo que podamos jerarquizar la información y definir los bloques de contenido.

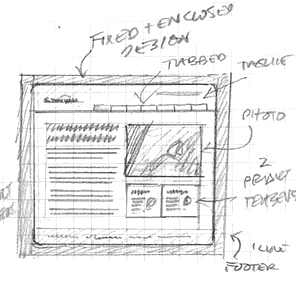
Para llevar a cabo este proceso tenemos varios pasos a seguir:

**Sketch -> Wireframe -> MockUp -> Prototipo interactivo -> Diseño final.**

## Sketch

Un sketch es un dibujo rápido o bosquejo de guía general que no tiene por qué tener muchos detalles y que reproduce un concepto, idea o generalidad de un proyecto de una manera muy sencilla. Por generalidad se realiza a mano con lápiz y borrador por su facilidad en el momento de plasmar una base o punto de partida.

**El entregable es una imagen o un papel.**



## Wireframe

Es una representación estática en baja calidad de un diseño. En esta representación se definen, para una fácil comprensión:

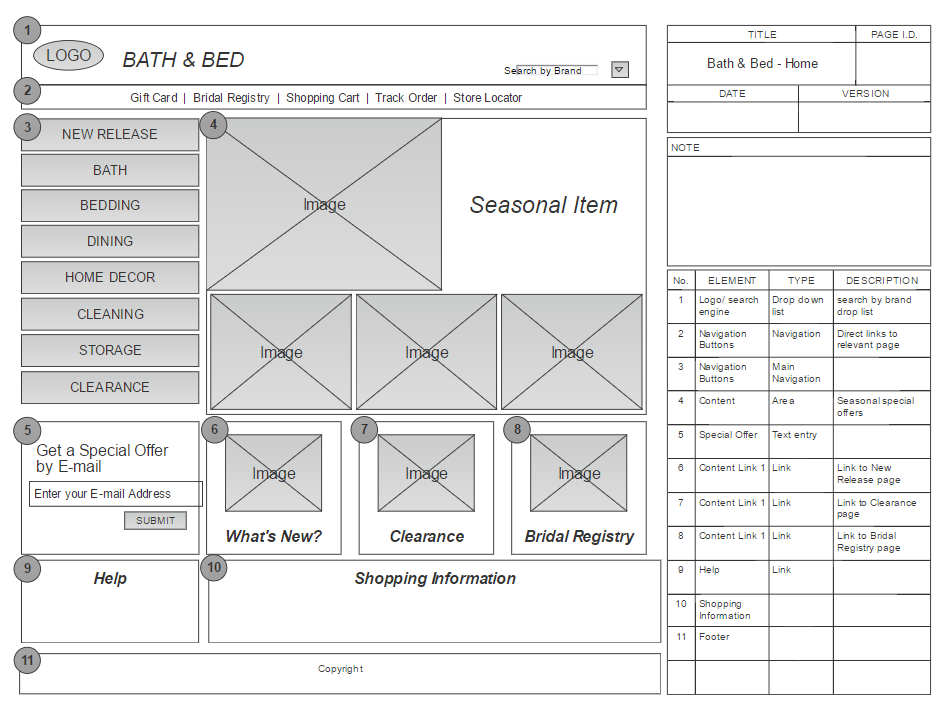
– **¿qué?** Los principales grupos de contenido

– **¿dónde?** La estructura de la información

– **¿cómo?** La descripción y visualización básica del usuario – interacción de la interfaz

Los Wireframes no son sólo un conjunto de cuadros grises, aunque puedan verse exactamente así. Considera a los Wireframes como la columna vertebral de tu diseño, donde tendrás una representación de cada parte importante del producto final.

**El entregable es una imagen.**



## Mockup

Representación estática de un diseño en calidad media o alta. Representa la estructura de la información, visualiza el contenido y demuestra las funcionalidades básicas de una manera estática. Permite revisar la parte visual real del proyecto.

**El entregable es una imagen.**

## Prototipo

Representación navegable del producto final. Simula la interacción interfaz-usuario y debe permitir que el usuario:

* Experimente interactuando con la interfaz y contenido del proyecto.
* Pruebe las principales interacciones de una manera similar al producto final.

**El entregable es un HTML, PPT, animación o formato navegable.**



# HTML

**Es un "lenguaje" de marcado (de etiquetas) para crear documentos para web**. Permite indicar dónde queremos cada elemento (párrafos, negritas, itálicas, imágenes, etc). Sólo se encarga de lo estructural, no del diseño (colores y tamaños son responsabilidad de CSS). Ha sido estandarizado por la W3C ([www.w3.org](http://www.w3.org)). Existen varias "versiones" (html4, xhtml, html5).

## Evolución del HTML

* HTML 1 y 2 (1991): Elementos de texto e imágenes.
* HTML 3 (1995): Basado en Tablas. No había un estándar: Cada browser interpretaba lo que quería.
* HTML 4 (1998): Aparece CSS, primer estándar oficial.
* XHTML 1.0 (2000): Es de la W3C, basado en XML.   
  TRANSITIONAL: Punto medio entre HTML4 y XHTML   
  STRICT: Más restrictivo (por ende más complejo).
* HTML5 (2004): Primer borrador creado por la WHATWG.   
  (2008) Pasa a la W3C.

**

## Otros lenguajes que complementan a HTML

**HTML no es el único lenguaje para hacer sitios web**. Entre los otros que existen se pueden mencionar:

* **CSS:** Aplica el formato visual y estructural de una página. Junto con HTML, pasan los diseños a documentos navegables.
* **Javascript:** Permite crear efectos-web (entre otras cualidades): Marquesinas, ventanas modales, drag and drop, validación de formularios. jQuery y AJAX son tecnologías basadas en Javascript.
* **PHP:** Para crear sitios dinámicos: Upload de archivos, envío de mails, conexión con la base de datos, registro y login de usuarios, sitios autoadministrables (con paneles de control).

WWW - La primera web nacida en Suiza en 1990. Proyecto homenaje realizado por **CERN**, el Centro Europeo de Física Nuclear, en Ginebra (Suiza) a 30 años de su creación. Podés navegarla acá <https://worldwideweb.cern.ch/browser/>

## Herramientas de desarrollo

* Cualquier programa que escriba texto plano (Notepad, Notepad++, Sublime Text, etc). **No sirven programas como Word** que llenan el archivo con «metadata» (información del software).
* Un navegador Web (o más de uno).

## Convenio de archivos

Los archivos de tu sitio web pueden llevar nombres arbitrarios, pero deben cumplir con tres reglas por convención:

1. No debe tener espacios, acentos, eñes, ni símbolos. Si son varias palabras usar guiones “ - “ o “ \_ “ ej: “mi-pagina-web”
2. Deben estar escritos en minúsculas.
3. Deben tener la extensión “.html” (esa es la forma que el servidor web sabe que es un documento web)

**ATENCIÓN** en Windows con la doble extensión de archivos (archivo.html.txt).

## Documento predeterminado

Es el archivo que se muestra cuando se pide una URL en un browser pero no se especifica un archivo. Dicho de otra manera: Es el archivo que se carga cuando todavía no hiciste click en ningún link. [ Al pedir [www.coderhouse.com](http://www.coderhouse.com), no dice qué archivo quiero ]

**Este archivo se debe llamar index.html ya que ese nombre es el estandarizado.**

## Etiquetas HTML

En el lenguaje HTML, los **elementos** se definen mediante **etiquetas**. Las etiquetas HTML indican el inicio y el final de cada elemento. Lo que se encuentra dentro de la etiqueta (el contenido) es lo que estás formateando.

Casi todas las etiquetas HTML son un juego de **pares**: una etiqueta abre, otra cierra. Para insertar una etiqueta se necesita escribir:

|  |
| --- |
| <etiqueta>  Contenido  </etiqueta> |

### Etiquetas cerradas y abiertas

Las cerradas encierran un contenido, por lo general texto. Las abiertas no encierran contenido, y sirven, entre otras cosas, para incluir elementos como imágenes, líneas, etc.

|  |
| --- |
| <p>Este es un párrafo con texto en su interior</p>  <hr/> |

En el ejemplo tenemos una etiqueta cerrada llamada **Párrafo** que engloba un texto y una etiqueta abierta para incluir una línea horizontal. El signo “/” se utiliza para las etiquetas de cierre; en las etiquetas cerradas se pone a continuación del signo “**<”**, en las abiertas se pone delante del signo “**>”**.

*En HTML5, ya no es una obligación tener que poner el signo “ / “, por ejemplo, <img src=””> funcionará correctamente, al igual que <br>, <hr>, o los meta tag.*

### Atributos de las etiquetas

Todas las etiquetas aceptan atributos. Un atributo es cualquier característica que puede ser diferente entre una etiqueta y la otra. Por ejemplo, en una foto, el ancho o alto.

|  |
| --- |
| <etiqueta atributo=”valor”>  Contenido  </etiqueta> |

El valor que tendrá va entre comillas. Una etiqueta puede tener más de un atributo, separados por espacios entre sí. Los atributos sólo van en la etiqueta de apertura.

### Anidar etiquetas

Se puede meter una etiqueta dentro de la otra (es más común de lo que parece). Lo más importante es tener presente que **siempre se cierran en orden inverso a la apertura**.



## Sintaxis del código

Al HTML no le importa si las etiquetas las escribís una al lado de la otra o una debajo de la otra. Los “Enter”, espacios (de la barra espaciadora) y tabulaciones **no afectan** la salida en el navegador. Este “espacio en blanco” se usa sólo para formatear el código de manera cómoda para el programador. Es una buena práctica que usen enter y tabulaciones para poder entender (de un primer vistazo) cómo está armada la estructura del sitio.

### Comentarios HTML

Es parte del código que el navegador no mostrará. Puede usarse para dejar una referencia de algo que se va a mostrar en algún lugar o para anular temporalmente un área del sitio. Tiene un inicio, un final y lo que está en el medio será omitido.

|  |
| --- |
| <!-- bloque comentado --> |

## Estructura básica

El HTML tiene una estructura de elementos que debe insertarse **siempre en cada documento**. Veamos cuáles son los elementos principales.

Cuando escribimos nuestro documento HTML, lo primero que tenemos que escribir es el DOCTYPE. **El doctype declara el tipo de documento**, es decir, nos sirve para indicar que nuestro documento está escrito siguiendo la estructura determinada por un DTD concreto.

**Un DTD es la definición de la estructura de un tipo de documento determinado.** Entonces, el DTD es dónde se define la estructura que debe tener el documento y utilizamos el doctype para informar qué DTD usamos.

### El doctype en HTML5

A partir de HTML5 la declaración del doctype es notablemente más sencilla, sólo basta con encabezar nuestro documento con la siguiente etiqueta.

|  |
| --- |
| <!DOCTYPE html> |

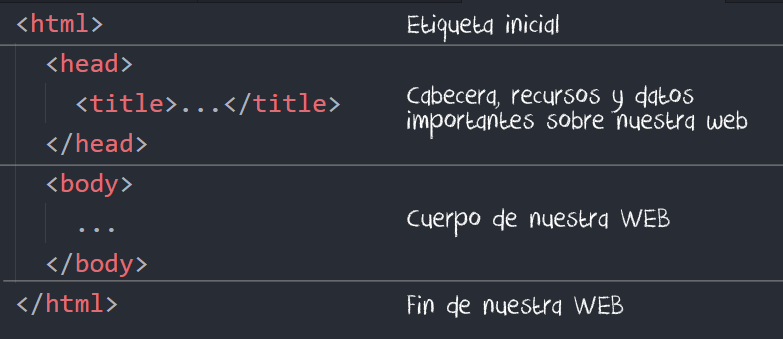
Una vez declarado el doctype, avanzamos con los elementos básicos.

* **<html>**: Etiqueta inicial, que define que el documento está bajo el estándar de HTML.

**Importante.** La etiqueta html abre y cierra. Es decir que todo el contenido del sitio tiene que estar entre <html> y </html>

En su interior anidaremos las siguientes etiquetas.

* **<head>**: Es la parte privada del documento, se utiliza como un espacio de comunicación entre el sitio web y el navegador. Esta etiqueta envuelve otras etiquetas importantes como **<title>,** las etiquetas **<meta>** y las relacionadas con vinculación de documentos CSS y JS.
* **<body>**: Encierra el contenido propiamente dicho del sitio.



**<title>:** La etiqueta title define el título de nuestra página, que será visualizado en la solapa del navegador.  
Como mencionamos también dentro del**<head>** puedes utilizar etiquetas **<meta>***:***<meta>:** Se utiliza para añadir información sobre la página, bien sea palabras clave, el autor, la descripción de nuestro sitio. Esta información puede ser utilizada por los buscadores. También puede definir el idioma y la codificación en la cual está escrita la página.

## Estructura básica completa

Utilizaremos todas estas etiquetas en nuestro primer ejemplo de estructura web:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <title>Mi primer sitio web</title>  </head>  <body>  ¡Este es mi primer sitio web!  </body>  </html> |

## Elementos HTML básicos

### Tipos de elementos: Grupo General

Todos los elementos que van en etiquetas dentro del **<body></body>** se dividen en dos grupos:

* **Elementos de bloque**: Son los que –sin ser modificados por CSS–, ocupan el 100% del ancho de su contenedor y se mostrarán uno abajo del otro.
* **Elementos de línea**: Solo ocupan el ancho que diga su contenido y se verán uno al lado del otro.

**<h1> a <h6> (de bloque):** Un encabezado es semánticamente el texto que “encabeza” (o titula) el contenido que sigue. Se puede tratar de un texto (como un artículo) o una sección del documento que estamos viendo.   
Existen 6 niveles: del **<h1></h1>** al **<h6></h6>**  
  
Niveles de encabezado:   
Estos niveles de encabezados no representan un texto en negrita más grande o más chico, detrás de estos “niveles” existe una jerarquía. No debés tener un **<h3>** si antes no hubo un **<h2>**. Además, están relacionados: Cada **<h3>** debe ser un sub-nivel de contenido del **<h2>** inmediato-anterior. Y si dos títulos son igual de importantes, son el mismo nivel de encabezado. Y si el título no es igual al anterior (ni tampoco es un sub-nivel) se volverá al nivel de encabezado anterior.  
Esta jerarquía se debe respetar **en cada documento HTML** que forme parte del sitio web.

|  |
| --- |
| <h1> REINO ANIMAL </h1>  <h2 Vertebrados </h2>  <h3> Mamiferos </h3>  <h4> Bipedos </h4>  <h4> Cuadrúpedos </h4>  <h3> Aves </h3>  <h4> Voladoras </h4>  <h4> No voladoras </h4>  <h3> Reptiles </h3>  <h3> Peces </h3>  <h3> Anfibios </h3>  <h2> Invertebrados </h2>  <h3> Insectos </h3>  <h4> Voladores </h4>  <h4> No Voladores </h4> |

**<p> (de bloque):** Un párrafo es un texto formado por una o más oraciones, pero sólo debe encerrar texto. Se inserta con la etiqueta **<p></p>**. Dentro de un párrafo se pueden insertar otros elementos de texto (negritas, itálicas, acrónimos, abreviaturas, vínculos).   
No puede haber un párrafo dentro de otro párrafo. Esto no existe en la literatura, y mucho menos en la estructura de un documento HTML.

**<br/> (de bloque):** Inserta un salto de línea simple. Para el HTML un Enter no significa nada. Para hacer un salto de línea se necesita una etiqueta. Ésta significa ***break*** y es para hacer un salto de línea dentro de un párrafo.  
**Importante: Son para separar oraciones, y no para separar un párrafo con otro**El **<br/>**es una etiqueta abierta, por lo tanto lleva la barra luego del nombre (o no lleva barra en absoluto).  
**<strong> (de línea):** El elemento strong es el apropiado para marcar con especial énfasis las partes más importantes de un texto. Visualmente, define el texto en Negrita.  
**<em> (de línea):**  Es la abreviatura de énfasis. Por ser un texto con énfasis, se leerá con un cambio significativo en el tono de voz. Visualmente, define el texto en *cursiva*.  
**<strong> (de línea):** Se trata de un texto reforzado (más fuerte) que será leído remarcando la palabra pero sin afectar al tono de voz. El elemento strong es el apropiado para marcar con especial énfasis las partes más importantes de un texto. Visualmente, define el texto en **negrita**.

### Elementos contenedores de usos múltiples

Existen dos etiquetas las cuales son utilizadas principalmente para dar formato CSS y orden -podríamos decir que son etiquetas "comodín"-. Estas son:

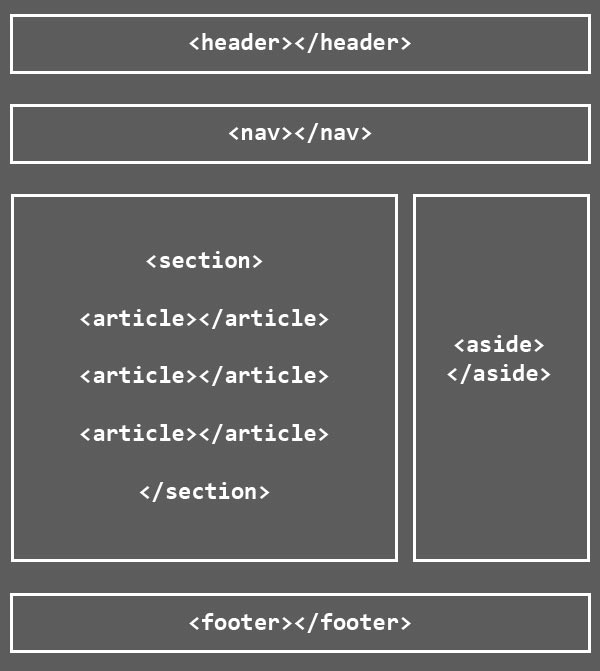
**<div> (de bloque):** Es un divisor (de ahí su nombre) y se comporta como elemento de bloque. Dentro del **<div></div>** puede ir cualquier elemento del HTML, incluso otros div’s. Es un elemento principal para armar maquetas con HTML y CSS. Como el **<div></div>** no produce una salida visual y acepta cualquier otro elemento dentro, se usa para delimitar áreas del sitio web.

**<span> (de línea):** Define un elemento en línea sin formato predefinido, solo debemos usarlo dentro de un texto en caso que se necesite cambiar el CSS.   
Por ejemplo; un encabezado a dos colores, sería un **<h1>**con un color para todo el texto y dentro un **<span>** con el texto de otro color.

*Con la aparición de las etiquetas semánticas, la utilización de estas etiquetas se ha disminuido un poco debido a que gran parte de su uso estaba principalmente enfocado en estructurar nuestros sitios web.*

## HTML5

HTML5 incorpora etiquetas semánticas que no sólo funcionan como contenedores, sino que también definen qué tipo de contenido van a tener.



### Nuevas etiquetas contenedoras HTML5

**<section>:** Define una sección en un documento.  
**<nav>:** Define un bloque que contiene enlaces de navegación, como por ejemplo el menú.  
**<article>:** Define contenido autónomo que podría existir independientemente del resto del contenido.  
**<aside>:** Define contenidos vagamente relacionados con el resto del contenido de la página.  
**<main>:** Define el contenido principal o importante en el documento. Solamente existe un elemento <main> en el documento.  
**<header>:**  Define la cabecera de una página o sección.  
**<footer>:** Define el pie de una página o sección.